

What is claimed is.

1 1. A computer implemented method of information re-
2 trieval in a file system, comprising the steps of:

3 displaying a portion of a hierarchical tree that is
4 representative of a repository of memorized files, levels
5 of said hierarchical tree comprising directories and sub-
6 directories thereunder; and

7 displaying a special virtual directory in each of
8 said directories and said subdirectories;

9 invoking a semantic operator by selection of said
10 special virtual directory;

11 displaying elements of at least a subtree of said hi-
12 erarchical tree, said elements being selected by said se-
13 mantic operator.

1 2. The method according to claim 1, wherein said step
2 of displaying further comprises arranging a screen dis-
3 play in accordance with a specification of said semantic
4 operator.

1 3. The method according to claim 1, wherein said ele-
2 ments comprise a portion of said directories.

1 4. The method according to claim 1, wherein said se-
2 mantic operator is _desc.

1 5. The method according to claim 1, wherein said se-
2 mantic operator is _star.

1 6. The method according to claim 1, wherein said re-
2 pository of memorized files comprises documents written
3 in a markup language.

1 7. A computer software product, comprising a com-
2 puter-readable medium in which computer program instruc-
3 tions are stored, which instructions, when read by a com-
4 puter, cause the computer to perform the steps of:
5 displaying a portion of a hierarchical tree that is
6 representative of a repository of memorized files, levels
7 of said hierarchical tree comprising directories and sub-
8 directories thereunder; and
9 displaying a special virtual directory in each of
10 said directories and said subdirectories;
11 invoking a semantic operator by selection of said
12 special virtual directory;
13 displaying elements of at least a subtree of said hi-
14 erarchical tree, said elements being selected by said se-
15 mantic operator.

1 8. The computer software product according to
2 claim 7, wherein said step of displaying further com-
3 prises arranging a screen display in accordance with a
4 specification of said semantic operator.

1 9. The computer software product according to
2 claim 7, wherein said elements comprise a portion of said
3 directories.

1 10. The computer software product according to
2 claim 7, wherein said semantic operator is _desc.

1 11. The computer software product according to
2 claim 7, wherein said semantic operator is _star.

1 12. The computer software product according to claim
2 7, wherein said repository of memorized files comprises
3 documents written in a markup language.

1 13. A computer implemented information retrieval sys-
2 tem for presenting a semantically dependent directory
3 structure of files to a user, comprising:

4 a file system engine, that receives a file request
5 via a file system application programming interface and
6 issues file system calls to an operating system, wherein
7 said file request specifies a file content of memorized
8 files,

9 wherein responsive to said file request, said file
10 system engine returns a hierarchical tree of directories
11 to said file system application programming interface,
12 said directories having references to selected ones of
13 said memorized files;

14 wherein said file system engine displays a special
15 virtual directory in each of said directories, wherein a

16 said special virtual directory comprises at least a por-
17 tion of said hierarchical tree, said portion being se-
18 lected by a semantic operator.

1 14. The information retrieval system according to
2 claim 13, further comprising a monitor, having a screen
3 display arranged thereon in accordance with a specifica-
4 tion of said semantic operator.

1 15. The information retrieval system according to
2 claim 13, wherein said semantic operator is _desc.

1 16. The information retrieval system according to
2 claim 13, wherein said semantic operator is _star.

1 17. The information retrieval system according to
2 claim 13, wherein said memorized files comprise documents
3 written in a markup language.

1 18. The information retrieval system according to
2 claim 17, wherein said markup language is XML.

1 19. A computer implemented method of information re-
2 trieval, comprising the steps of:

3 retrieving structural information of memorized docu-
4 ments according to a document type declaration that cor-
5 responds to each of said documents;

6 retrieving elements, attributes and values of said
7 elements and said attributes of said documents;

8 generating a multilevel inverted index from said
9 structural information, said elements, said attributes
10 and said values;

11 accepting a specification from a user having members
12 that comprise at least one of said elements, said attrib-
13 utes and said values;

14 responsive to said specification, extracting data
15 from said multilevel inverted index that complies with at
16 least one of said members;

17 displaying a hierarchical tree, levels of said hier-
18 archical tree comprising directories, wherein said direc-
19 tories each comprise a sequence of said members, and
20 wherein contents of said directories and contents of sub-
21 directories thereunder comprise selected ones of said
22 documents possessing said specification; and

23 displaying a special virtual directory in each of
24 said directories, wherein a content of said special vir-
25 tual directory comprises at least one level of said hier-
26 archical tree, said one level being more deeply nested
27 than a level of said special virtual directory in said
28 hierarchical tree.

1 20. The method according to claim 19, wherein said
2 step of displaying said special virtual directory com-
3 prises invoking an operator _desc to a context node of
4 said special virtual directory.

1 21. The method according to claim 20, wherein said
2 step of invoking said operator _desc further comprises
3 the steps of:

4 selecting all descendants of said context node; and
5 displaying a list of said descendants.

1 22. The method according to claim 21, wherein said
2 list is a linear list.

1 23. The method according to claim 19, wherein said
2 step of displaying said special virtual directory com-
3 prises invoking an operator _star to a context node of
4 said special virtual directory.

1 24. The method according to claim 23, wherein said
2 step of invoking said operator _star further comprises
3 the steps of:

4 selecting all children of said context node; and dis-
5 playing a list of grandchildren of said context node.

1 25. The method according to claim 24, wherein said
2 list is a linear list.

1 26. The method according to claim 24, wherein said
2 children are selected from said elements.

1 27. The method according to claim 24, wherein said
2 children comprise selected ones of said elements, said
3 attributes, and said values.

1 28. The method according to claim 19, wherein said
2 multilevel inverted index comprises a structural section
3 having postings of said structural information, and a
4 words section having postings of said values, wherein
5 said values are words.

1 29. The method according to claim 19, wherein said
2 documents are XML documents.

1 30. The method according to claim 19, further com-
2 prising the steps of:

3 noting changes in a composition of a repository of
4 said documents; and

5 updating said multilevel inverted index responsive to
6 said changes.

1 31. A computer implemented method of information re-
2 trieval, comprising the steps of:

3 retrieving structural information of memorized docu-
4 ments according to a document type declaration that cor-
5 responds to each of said documents, wherein said docu-
6 ments are written in a markup language;

7 retrieving elements, attributes and values of said
8 elements and said attributes of said documents;

9 generating a multilevel inverted index from said
10 structural information, said elements, said attributes
11 and said values;

12 accepting a specification from a user having members
13 that comprise at least one of said elements, said attrib-
14 utes and said values;

15 responsive to said specification, extracting data
16 from said multilevel inverted index that complies with at
17 least one of said members;

18 displaying a hierarchical tree, levels of said hier-
19 archical tree comprising directories, wherein said direc-
20 tories each comprise a sequence of said members, and
21 wherein contents of said directories and contents of sub-
22 directories thereunder comprise selected ones of said
23 documents possessing said specification; and

24 displaying a special virtual directory in each of
25 said directories, wherein a content of said special vir-
26 tual directory comprises at least one level of said hier-
27 archical tree, said one level being more deeply nested
28 than a level of said special virtual directory in said
29 hierarchical tree.

1 32. The method according to claim 31, wherein said
2 step of displaying said special virtual directory com-
3 prises invoking an operator _desc to a context node of
4 said special virtual directory.

1 33. The method according to claim 32, wherein said
2 step of invoking said operator _desc further comprises
3 the steps of:

4 selecting all descendants of said context node; and
5 displaying a list of said descendants.

1 34. The method according to claim 33, wherein said
2 list is a linear list.

1 35. The method according to claim 31, wherein said
2 step of displaying said special virtual directory com-
3 prises invoking an operator _star to a context node of
4 said special virtual directory.

1 36. The method according to claim 35, wherein said
2 step of invoking said operator _star further comprises
3 the steps of:

4 selecting all children of said context node; and dis-
5 playing a list of grandchildren of said context node.

1 37. The method according to claim 36, wherein said
2 list is a linear list.

1 38. The method according to claim 36, wherein said
2 children are selected from said elements.

1 39. The method according to claim 36, wherein said
2 children comprise selected ones of said elements, said
3 attributes, and said values.

1 40. The method according to claim 31, wherein said
2 multilevel inverted index comprises a structural section
3 having postings of said structural information, and a
4 words section having postings of said values, wherein
5 said values are words.

1 41. The method according to claim 31, wherein said
2 documents are XML documents.

1 42. The method according to claim 31, further com-
2 prising the steps of:

3 noting changes in a composition of a repository of
4 said documents; and

5 updating said multilevel inverted index responsive to
6 said changes.

1 43. A computer software product, comprising a com-
2 puter-readable medium in which computer program instruc-
3 tions are stored, which instructions, when read by a com-
4 puter, cause the computer to perform the steps of:

5 retrieving structural information of memorized docu-
6 ments according to a document type declaration that cor-
7 responds to each of said documents;

8 retrieving elements, attributes and values of said
9 elements and said attributes of said documents;

10 generating a multilevel inverted index from said
11 structural information, said elements, said attributes
12 and said values;

13 accepting a specification from a user having members
14 that comprise at least one of said elements, said attrib-
15 utes and said values;

16 responsive to said specification, extracting data
17 from said multilevel inverted index that complies with at
18 least one of said members; -

19 displaying a hierarchical tree, levels of said hier-
20 archical tree comprising directories, wherein said direc-
21 tories each comprise a sequence of said members, and
22 wherein contents of said directories and contents of sub-
23 directories thereunder comprise selected ones of said
24 documents possessing said specification; and

25 displaying a special virtual directory in each of
26 said directories, wherein a content of said special vir-
27 tual directory comprises at least one level of said hier-
28 archical tree, said one level being more deeply nested
29 than a level of said special virtual directory in said
30 hierarchical tree.

1 44. The computer software product according to
2 claim 43, wherein said step of displaying said special
3 virtual directory comprises invoking an operator _desc to
4 a context node of said special virtual directory.

1 45. The computer software product according to
2 claim 44, wherein said step of invoking said operator
3 _desc further comprises the steps of:

4 selecting all descendants of said context node; and
5 displaying a list of said descendants.

1 46. The computer software product according to
2 claim 45, wherein said list is a linear list.

1 47. The computer software product according to
2 claim 43, wherein said step of invoking said operator
3 _star comprises invoking an operator _star to a context
4 node of said special virtual directory.

1 48. The computer software product according to
2 claim 47, wherein said step of invoking said operator
3 _star further comprises the steps of:

4 selecting all children of said context node; and dis-
5 playing a list of grandchildren of said context node.

1 49. The computer software product according to
2 claim 48, wherein said list is a linear list.

1 50. The computer software product according to
2 claim 48, wherein said children are selected from said
3 elements.

1 51. The computer software product according to
2 claim 48, wherein said children comprise selected ones of
3 said elements, said attributes, and said values.

1 52. The computer software product according to
2 claim 43, wherein said multilevel inverted index com-
3 prises a structural section having postings of said
4 structural information, and a words section having post-
5 ings of said values, wherein said values are words.

1 53. The computer software product according to
2 claim 43, wherein said documents are XML documents.

1 54. The computer software product according to
2 claim 43, wherein said instructions further cause the
3 computer to perform the steps of:

4 noting changes in a composition of a repository of
5 said documents; and

6 updating said multilevel inverted index responsive to
7 said changes.

1 55. A computer software product, comprising a com-
2 puter-readable medium in which computer program instruc-
3 tions are stored, which instructions, when read by a com-
4 puter, cause the computer to perform the steps of:

5 retrieving structural information of memorized docu-
6 ments according to a document type declaration that cor-
7 responds to each of said documents, wherein said docu-
8 ments are written in a markup language;

9 retrieving elements, attributes and values of said
10 elements and said attributes of said documents;

11 generating a multilevel inverted index from said
12 structural information, said elements, said attributes
13 and said values;

14 accepting a specification from a user having members
15 that comprise at least one of said elements, said attrib-
16 utes and said values;

17 responsive to said specification, extracting data
18 from said multilevel inverted index that complies with at
19 least one of said members;

20 displaying a hierarchical tree, levels of said hier-
21 archical tree comprising virtual directories, wherein
22 said virtual directories each comprise a sequence of said
23 members, and wherein contents of said virtual directories
24 and contents of virtual subdirectories thereunder com-
25 prise selected ones of said documents possessing said
26 specification; and

27 displaying a special virtual directory in each of
28 said virtual directories, wherein a content of said spe-
29 cial virtual directory comprises at least one level of
30 said hierarchical tree, said one level being more deeply
31 nested than a level of said special virtual directory in
32 said hierarchical tree.

1 56. The computer software product according to
2 claim 55, wherein said step of displaying said special
3 virtual directory comprises invoking an operator _desc to
4 a context node of said special virtual directory.

1 57. The computer software product according to
2 claim 56, wherein said step of invoking said operator
3 _desc further comprises the steps of:

4 selecting all descendants of said context node; and
5 displaying a list of said descendants.

1 58. The computer software product according to
2 claim 57, wherein said list is a linear list.

1 59. The computer software product according to
2 claim 55, wherein said step of displaying said special
3 virtual directory comprises invoking an operator _star to
4 a context node of said special virtual directory.

1 60. The computer software product according to
2 claim 59, wherein said step of invoking said operator
3 _star further comprises the steps of:

4 selecting all children of said context node; and dis-
5 playing a list of grandchildren of said context node.

1 61. The computer software product according to
2 claim 60, wherein said list is a linear list.

1 62. The computer software product according to
2 claim 60, wherein said children are selected from said
3 elements.

1 63. The computer software product according to
2 claim 60, wherein said children comprise selected ones of
3 said elements, said attributes, and said values.

1 64. The computer software product according to
2 claim 55, wherein said multilevel inverted index com-
3 prises a structural section having postings of said
4 structural information, and a words section having post-
5 ings of said values, wherein said values are words.

1 65. The computer software product according to
2 claim 55, wherein said documents are XML documents.

1 66. The computer software product according to
2 claim 55, wherein said instructions further cause the
3 computer to perform the steps of:

4 noting changes in a composition of a repository of
5 said documents; and

6 updating said multilevel inverted index responsive to
7 said changes.

1 67. A computer implemented information retrieval sys-
2 tem for presenting a semantically dependent directory
3 structure of files to a user, comprising:

4 a file system engine, that receives a file request
5 via a file system application programming interface and
6 issues file system calls to an operating system, wherein
7 said file request specifies a file content of memorized

8 files, wherein said files comprise documents written in a
9 markup language;

10 a parser, linked to said file system engine, that re-
11 trieves structural information of said documents, said
12 parser further retrieving at least one of elements, at-
13 tributes and respective values thereof from said docu-
14 ments;

15 an indexer, linked to said parser, for constructing
16 an inverted index of said elements and said attributes
17 and said respective values thereof,

18 wherein responsive to said file request, said file
19 system engine retrieves postings of said inverted index
20 that satisfy requirements of said file request, and re-
21 turns a hierarchical tree of directories to said file
22 system application programming interface, said directo-
23 ries having references to selected ones of said documents
24 corresponding to said postings;

25 wherein said file system engine displays a special
26 virtual directory in each of said directories, wherein a
27 content of said special virtual directory comprises at
28 least one level of said hierarchical tree, said one level
29 being more deeply nested than a level of said special
30 virtual directory in said hierarchical tree.

1 68. The information retrieval system according to
2 claim 67, wherein said file system engine displays said
3 special virtual directory by invoking an operator _desc
4 to a context node of said special virtual directory.

1 69. The information retrieval system according to
2 claim 68, wherein said file system engine displays said
3 special virtual directory by the steps of:

4 selecting all descendants of said context node; and
5 displaying a list of said descendants.

1 70. The information retrieval system according to
2 claim 69, wherein said list is a linear list.

1 71. The information retrieval system according to
2 claim 67, wherein said file system engine displays said
3 special virtual directory by invoking an operator _star
4 to a context node of said special virtual directory.

1 72. The information retrieval system according to
2 claim 67, wherein said file system engine displays said
3 special virtual directory by the steps of:

4 selecting all children of a context node of said spe-
5 cial virtual directory; and displaying a list of grand-
6 children of said context node.

1 73. The information retrieval system according to
2 claim 72, wherein said list is a linear list.

1 74. The information retrieval system according to
2 claim 72, wherein said children are selected from said
3 elements.

1 75. The information retrieval system according to
2 claim 72, wherein said children comprise selected ones of
3 said elements, said attributes, and said values.

1 76. The information retrieval system of claim 67,
2 wherein said inverted index comprises a structural sec-
3 tion having postings of said structural information, and
4 a words section having postings of words of said docu-
5 ments.

1 77. The information retrieval system of claim 67,
2 further comprising an analyzer for updating said inverted
3 index, wherein said analyzer analyzes additions to said
4 memorized files.

1 78. The information retrieval system of claim 67,
2 wherein said parser retrieves said structural information
3 from document type declarations of said documents.

1 79. A computer implemented information retrieval sys-
2 tem for presenting a semantically dependent directory
3 structure of XML files to a user, comprising:

4 a file system engine, that receives a file request
5 via a file system application programming interface and
6 issues file system calls to an operating system, wherein
7 said file request specifies a file content of memorized
8 files;

9 an XML parser, linked to said file system engine,
10 that retrieves structural information of XML documents,
11 said XML parser further retrieving at least one of ele-
12 ments, attributes and respective values thereof from said
13 XML documents;

14 an indexer, linked to said XML parser, for construct-
15 ing an inverted index of said elements and said attrib-
16 utes and said respective values thereof,

17 wherein responsive to said file request, said file
18 system engine retrieves postings of said inverted index
19 that satisfy requirements of said file request, and re-
20 turns a hierarchical tree of virtual directories to said
21 file system application programming interface, said vir-
22 tual directories having references to selected ones of
23 said XML documents corresponding to said postings;

24 wherein said file system engine displays a special
25 virtual directory in each of said virtual directories,
26 wherein a content of said special virtual directory com-
27 prises at least one level of said hierarchical tree, said
28 one level being more deeply nested than a level of said
29 special virtual directory in said hierarchical tree.

1 80. The information retrieval system according to
2 claim 79, wherein said file system engine displays said
3 special virtual directory by invoking an operator _desc
4 to a context node of said special virtual directory.

1 81. The information retrieval system according to
2 claim 80, wherein said file system engine displays said
3 special virtual directory by the steps of:

4 selecting all descendants of said context node; and
5 displaying a list of said descendants.

1 82. The information retrieval system according to
2 claim 81, wherein said list is a linear list.

1 83. The information retrieval system according to
2 claim 79, wherein said file system engine displays said
3 special virtual directory by invoking an operator _star
4 to a context node of said special virtual directory.

1 84. The information retrieval system according to
2 claim 83, wherein said file system engine displays said
3 special virtual directory by the steps of:

4 selecting all children of said context node of said
5 special virtual directory; and displaying a list of
6 grandchildren of said context node.

1 85. The information retrieval system according to
2 claim 84, wherein said list is a linear list.

1 86. The information retrieval system according to
2 claim 84, wherein said children are selected from said
3 elements.

1 87. The information retrieval system according to
2 claim 84, wherein said children comprise selected ones of
3 said elements, said attributes, and said respective val-
4 ues.

1 88. The information retrieval system of claim 79,
2 wherein said inverted index comprises a structural sec-
3 tion having postings of said structural information, and
4 a words section having postings of words of said XML
5 documents.

1 89. The information retrieval system of claim 79,
2 further comprising an XML analyzer for updating said in-
3 verted index, wherein said XML analyzer analyzes addi-
4 tions to said memorized files.

1 90. The information retrieval system of claim 79,
2 wherein said XML parser retrieves said structural infor-
3 mation from document type declarations of said XML docu-
4 ments.